

WAVEFORM DATA EDITING DEVICE

Patent number: JP6175663
Publication date: 1994-06-24
Inventor: ADACHI ATSUSHI
Applicant: YAMAHA CORP
Classification:
- international: G10H7/00; G11B27/034
- european:
Application number: JP19920323453 19921202
Priority number(s):

Abstract of JP6175663

PURPOSE: To provide a waveform data editing device which facilitates editing operation for waveform data generated in consideration of the length of a note.

CONSTITUTION: In a step Sa3, arithmetic operation is performed on the basis of the sampling rate at the time of waveform data recording, a tempo obtained by setting or calculation, a head sample position which is inputted in a step Sa1, and note length which is inputted in a step Sa2 to calculate a final sample position corresponding to the tail position in the range of a waveform to be edited. Consequently, the range in which the waveform data are to be edited can be specified with the note length.

先頭サン

音

サンプリング
先頭サンプル
に基づき最終
を算出

表示
処

Data supplied from the esp@cenet database - Worldwide

(19)日本国特許庁(J P)

(12) 公開特許公報(A)

(11)特許出願公開番号

特開平6-175663

(43)公開日 平成6年(1994)6月24日

(51)Int.Cl. ⁵	識別記号	庁内整理番号	F I	技術表示箇所
G 1 0 H 7/00		8622-5H		
G 1 1 B 27/034		8224-5D	G 1 1 B 27/ 02	H
		8622-5H	G 1 0 H 7/ 00	5 1 1 J

審査請求 未請求 請求項の数2(全 7 頁)

(21)出願番号 特願平4-323453

(22)出願日 平成4年(1992)12月2日

(71)出願人 000004075

ヤマハ株式会社

静岡県浜松市中沢町10番1号

(72)発明者 安達 淳

静岡県浜松市中沢町10番1号 ヤマハ株式会社内

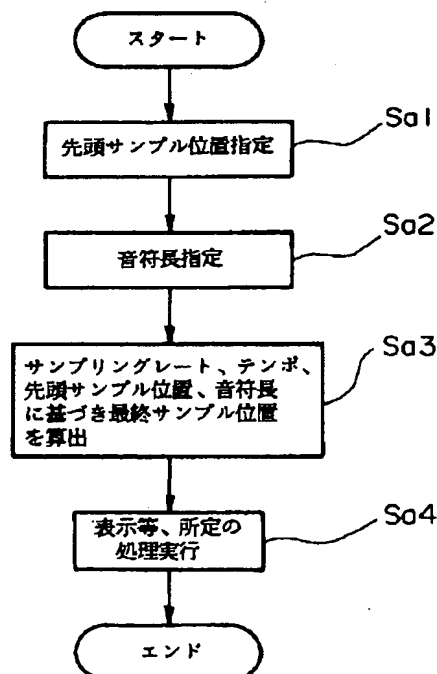
(74)代理人 弁理士 志賀 正武 (外2名)

(54)【発明の名称】 波形データ編集装置

(57)【要約】

【目的】 音符の長さを意識した波形データの編集作業を容易に行うことができる波形データ編集装置を提供する。

【構成】 ステップS a 3では、波形データ記録時のサンプリングレート、設定あるいは算出により得られるテンポ、ステップS a 1で入力された先頭サンプル位置、およびステップS a 2で入力された音符長に基づいて演算を行い、編集すべき波形の範囲の最終位置に対応する最終サンプル位置を算出するようにした。これにより、波形データの編集すべき範囲を音符長により指定することができる。



【特許請求の範囲】

【請求項1】 所定の演奏速度で演奏された楽音の波形データを所定のサンプリング周波数で記録し、この記録した波形データのうち所望の範囲を指定して該範囲内の波形データに対して編集処理を施す波形データ編集装置において、

前記範囲の開始位置を指定する指定手段と、

所望の音符長を入力する入力手段と、

前記サンプリング周波数、前記演奏速度、前記指定された開始位置および前記入力された音符長に基づき、前記範囲の終了位置を算出する演算手段とを具備することを特徴とする波形データ編集装置。

【請求項2】 所定の演奏速度で演奏された楽音の波形データを所定のサンプリング周波数で記録し、この記録した波形データのうち所望の範囲を指定して該範囲内の波形データに対して編集処理を施す波形データ編集装置において、

前記範囲の開始位置を指定する第1の指定手段と、

前記範囲の終了位置を指定する第2の指定手段と、

前記サンプリング周波数、前記演奏速度、前記指定された開始位置および前記指定された終了位置に基づき、前記範囲に相当する音符長を算出する演算手段とを具備することを特徴とする波形データ編集装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 この発明は、例えば演奏収録により得られた楽音の波形データを容易に編集することができる波形データ編集装置に関する。

【0002】

【従来の技術】 ハードディスク等の記憶媒体にデジタルオーディオデータを記録するデジタルレコーダが知られている。一般に、デジタルレコーダは、記録した波形データを編集する機能を有しており、例えば波形データの一部を切りとって、これを他の部分にコピーするなど種々の編集処理を行うことができる。このようなデジタルレコーダにおいて編集すべき波形データの範囲を指定する場合、まず記録した波形データを表す波形を表示装置に表示させ、操作者がテンキーやデータエントリーダイヤル等を用いて指定すべき範囲の先頭位置と最終位置とを入力するようになっている。

【0003】

【発明が解決しようとする課題】 ところで、上記従来のデジタルレコーダでは、波形の位置を単に先頭からのサンプル位置等によって指示するようになっていたもので、テンポや拍子等を有する音楽の波形データを編集する場合、音符の長さを意識して波形位置を指示することが容易でなかった。このため、デジタルレコーダを用いた音楽の波形データの編集作業に多大な労力が必要となるという問題があった。

【0004】 この発明は、このような背景の下になされ

たもので、音符の長さを意識した波形データの編集作業を容易に行うことができる波形データ編集装置を提供することを目的としている。

【0005】

【課題を解決するための手段】 上述した課題を解決するために、請求項1記載の発明は、所定の演奏速度で演奏された楽音の波形データを所定のサンプリング周波数で記録し、この記録した波形データのうち所望の範囲を指定して該範囲内の波形データに対して編集処理を施す波形データ編集装置において、前記範囲の開始位置を指定する指定手段と、所望の音符長を入力する入力手段と、前記サンプリング周波数、前記演奏速度、前記指定された開始位置および前記入力された音符長に基づき、前記範囲の終了位置を算出する演算手段とを具備することを特徴としている。

【0006】 また、請求項2記載の発明は、所定の演奏速度で演奏された楽音の波形データを所定のサンプリング周波数で記録し、この記録した波形データのうち所望の範囲を指定して該範囲内の波形データに対して編集処理を施す波形データ編集装置において、前記範囲の開始位置を指定する第1の指定手段と、前記範囲の終了位置を指定する第2の指定手段と、前記サンプリング周波数、前記演奏速度、前記指定された開始位置および前記指定された終了位置に基づき、前記範囲に相当する音符長を算出する演算手段とを具備することを特徴としている。

【0007】

【作用】 請求項1記載の発明によれば、指定手段が、編集処理を施すべき所望の範囲の開始位置を指定し、入力手段が、所望の音符長を入力し、演算手段が、演奏記録時のサンプリング周波数、演奏速度、前記指定された開始位置および前記入力された音符長に基づき、前記範囲の終了位置を算出する。これにより、編集処理を施すべき波形データの範囲を音符長で指定することができる。

【0008】 また、請求項2記載の発明によれば、第1の指定手段が、編集処理を施すべき所望の範囲の開始位置を指定し、第2の指定手段が、該範囲の終了位置を指定し、演算手段が、演奏記録時のサンプリング周波数、演奏速度、前記指定された開始位置および前記指定された終了位置に基づき、前記範囲に相当する音符長を算出する。これにより、編集処理を施すべき波形データの範囲を指定する際に、該範囲の長さを音符長で知ることができる。

【0009】

【実施例】 以下、図面を参照して、この発明の実施例について説明する。図1はこの発明の第1実施例の全体構成を示すブロック図である。この図において、1はバスを介して接続される装置各部を制御するCPU（中央処理装置）であり、その動作については後述する。2はROM（Read Only Memory）であり、CPU1によってロ

3

ードされる各種制御プログラムが記憶される。3はRAM (Random Access Memory) であり、CPU1のワークエリアとして用いられ、各種レジスタ値や演算結果が記憶される。また、このRAM3は、楽音波形データを記憶する波形メモリとしても用いられる。さらに、4はハードディスク等を記憶媒体とする外部記憶装置であり、RAM3からバスを介し転送される楽音波形データが格納される。

【0010】5はA/D (アナログ/デジタル) 変換器であり、外部から供給されるアナログの楽音信号をデジタルの楽音波形データに変換し、これをバスを介しRAM3へ転送する。6はスイッチ群であり、数値入力を行うテンキー、入力値に「±1」の増減を行うプラス/マイナスキー、音符長を入力する音符キー等の各種キースイッチから構成されている。7はスイッチ検出回路であり、スイッチ群6の各キースイッチに対する操作を検出し、これに応じた操作信号を発生する。

【0011】また、8は表示回路であり、RAM3から供給される楽音波形データを表示用データに変換し、その波形をモニタに表示させる。ここで、図2は、ドラムを演奏した場合の楽音波形をモニタに表示した例を示している。この図において、横軸は時間(秒)を、縦軸は波形の振幅を示している。なお、横軸の時間は、記録開始時点からの絶対時間あるいは操作者が任意に指定した時点からの相対時間の何れによって表現することも可能である。また、表示画面の右上には、図示のように記録した楽曲のテンポが表示される。この場合、テンポは、例えば下記第1～第3の方法によって求めることができる。

【0012】① 第1の方法は、記録した楽曲が例えば4/4拍子で120小節あり、演奏時間が4分30秒である場合、これらの値から計算によってテンポ=106.67を得る方法である。

② 第2の方法は、記録した楽曲の波形データから拍のタイミングを抽出し、抽出した拍のタイミングの間隔からテンポを算出する方法である。この方法によれば、曲の途中でテンポが変化する場合にも対応できるという利点がある。なお、この方法の詳細については、例えば特開昭62-254191号に開示されている。

③ 第3の方法は、電子楽器の自動演奏とともに楽曲を記録する方法である。すなわち、自動演奏において再生される演奏情報にはテンポに関するデータが含まれているので、このデータを用いれば容易にテンポを求めることができる。

なお、上記第1～第3の方法によらずに操作者が所望のテンポの値を任意に指定するようにしてもよい。

【0013】また、図2に示した波形のグラフは、図3に示すように横軸の時間を小節番号と該小節の先頭から

とすると、最終サンプル位置-先頭サンプル位置=44100×(60/120
) ×(4×2/4

4

の拍数(図中4-2, 4-3, ……、5-2)とによって表現することも可能である。

【0014】さて、再び図1に戻って実施例の構成を説明する。同図において、9は読み出し回路であり、RAM3に記憶された楽音波形データを指定された所定のアドレスから所定のレートで読み出し、これを出力する。10はD/A (デジタル/アナログ) 変換器であり、読み出し回路9から供給されるデジタルの楽音波形データをアナログの楽音信号に変換し、図示しないサウンドシステム等の外部装置へ出力する。

【0015】次に、上記構成による実施例の動作を説明する。まず、電源が投入されると、CPU1はROM2に記憶された制御プログラムをロードし、これを実行する。そして、外部から楽音信号が入力されると、これがA/D変換器5においてデジタルの楽音波形データに変換され、RAM3に記憶される。

【0016】そして、操作者が編集開始を指示するスイッチをオンすると、CPU1はロードした制御プログラムのうち図4に示す編集処理ルーチンを起動する。これにより、CPU1の処理がこのルーチンのステップSa1に進む。ステップSa1では、RAM3に記憶された楽音波形データが表示回路8に供給され、表示用データに変換される。これにより、図2に示した波形がモニタに表示される。

【0017】ここで、操作者は、テンキーによる数値入力、あるいはデータエントリダイヤルの操作によるカーソル移動などにより、編集すべき波形の先頭位置を波形データの先頭サンプルからのサンプル数(以下、先頭サンプル位置と称する。)によって指定する。これにより、指定された先頭サンプル位置(図示略)がモニタに表示される。

【0018】次に、ステップSa2では、操作者が編集すべき波形の先頭サンプル位置からの範囲を、音符キーを用いて音符長(例えば4分音符の2倍など)によって入力する。これにより、編集すべき波形の先頭サンプル位置からの範囲が特定されることになる。そして、ステップSa3に進むと、波形記録時のサンプリングレート、設定あるいは算出により得られるテンポ、上記ステップSa1で入力された先頭サンプル位置、および上記ステップSa2で入力された音符長に基づいて演算を行い、編集すべき波形の範囲の最終位置に対応するサンプル位置(以下、最終サンプル位置と称する。)を算出する。

【0019】例えば、サンプリング周波数=44.1kHz、
テンポ=120、
音符長=(4分音符)×2

となる。したがって、最終サンプル位置は先頭サンプル位置にサンプル数「44100」を加えた値となる。

【0020】次に、ステップS a 4に進むと、図5に示すように、上記ステップS a 3の演算によって得られた編集すべき波形の範囲Rを表示する。同図において、範囲Rは、先頭サンプル位置S Pから4分音符の2倍の音符長後に相当する最終サンプル位置E Pまでとなっている。そして、操作者の指示に応じて下記①～③に示す編集処理を行う。

【0021】①削除処理

操作者が所定のスイッチを操作して波形の削除を指示すると、図6に示すように、範囲R内にある波形の振幅が「0」となる。こうして、範囲R内の波形が削除される。また、この場合、範囲Rに対応する波形データの振幅を実際に「0」にするのではなく、楽音再生時にこの範囲R内の波形データを再生しないよう削除フラグを反転させる等、ソフトウェア的に制御することも可能である。このようにすれば、過去に削除を指示した波形データを後に復活させることが可能となる。こうして、例えば不良な波形を削除することにより、再度この削除した部分について波形を記録し直せば、不良な部分を適宜良好な波形に修正することができる。

【0022】②移動処理

また、操作者が所定のスイッチを操作して波形の移動を指示し、所望の移動先を入力すると、範囲R内にある波形が例えば図7に示すように移動する。この図から分かるように、波形の移動によって元の波形がなくなった部分Eについては振幅が「0」となり、移動先の部分Cについては移動される範囲Rの波形が書き込まれる。このような移動処理によって例えば波形の位置をずらすことにより、発音タイミングの補正等を容易に行うことができる。また、この場合も上記削除処理と同様、実際にはデータを動かさず、ソフトウェア的に制御してもよい。

【0023】③コピー処理

また、操作者が所定のスイッチを操作して波形のコピーを指示し、所望のコピー先を入力すると、範囲R内にある波形が例えば図8に示すようにコピー先Tに書き込まれる。こうすることにより、例えば良好な状態で記録された所定の波形を他の部分に複写して繰り返し利用したり、ある短い部分（例えば1拍）を複数連続してコピーすることにより所定長（例えば1小節）のフレーズを作成したりすることが可能になる。また、この場合も上記削除処理や移動処理と同様、実際には複数の同じデータを持つようにするのではなく、ソフトウェア的に制御してもよい。

【0024】さらに、上記指定された範囲R（図5参照）を以下の用途④～⑦に利用することも可能である。

④単に、モニタに表示されている最終サンプル位置E P

= 44100 (サンプル数)

を波形編集時の目安にする。

⑤範囲Rにおいてパンチイン録音を行う。

⑥最終サンプル位置E Pを再生開始アドレスにする。これにより、例えば演奏開始時にテンポをとるためにカウント打ちを行った場合、この部分を再生しないようにすることが可能になる。

⑦先頭サンプル位置S Pを移動することにより範囲Rを所望の位置に平行移動させた後、上記①～⑤の処理を行う。

【0025】このように、本実施例によれば、編集すべき波形の範囲を所望の時点から音符長単位で指定することができるので、収録した音楽の波形データを編集する作業を容易に行うことができる。

【0026】次に、この発明の第2実施例について説明する。この実施例が上記第1実施例と異なる点は、編集すべき波形の範囲を指定する方法にある。この実施例のハードウェア構成は、図1に示したものと同様であるので、同一の符号を使用し、その説明を省略する。

【0027】以下、図9に示すフローチャートを参照し、この実施例の動作を説明する。電源投入後、操作者が編集開始を指示すると、この実施例の編集処理ルーチンが起動され、CPU1の処理はステップS b 1（図9参照）に進む。

【0028】ステップS b 1では、上記第1実施例と同様、編集を施すべき波形（図2参照）がモニタに表示される。ここでは、操作者が編集すべき範囲の先頭サンプル位置を入力する。そして、ステップS b 2に進むと、操作者が編集すべき波形の最終サンプル位置を入力する。これにより、編集すべき波形の範囲が特定されることになる。

【0029】次に、ステップS b 3に進むと、波形記録時のサンプリングレート、設定あるいは算出により得られるテンポ、上記ステップS b 1で入力された先頭サンプル位置、および上記ステップS b 2で入力された最終サンプル位置に基づいて演算を行い、編集すべき波形の範囲に相当する音符長を算出する。

【0030】例えば、サンプリング周波数=44.1kHz、

テンポ=120、

最終サンプル位置-先頭サンプル位置=84500 (サンプル数)

とすると、4分音符の長さを基準として計算した場合、音符長=3.832

= (4分音符) × 3 + (符点8分音符) × 1 + α (余り)

となる。そして、この算出された音符長をモニタに表示させる。

【0031】ここで、操作者は、何度でも上記ステップ

7

S b 2における最終サンプル位置の入力をし直すことが可能であり、この入力のし直しがある度に上記ステップS b 3において音符長を再計算する。図10は、最終サンプル位置E Pの入力のし直しにより、編集すべき波形の範囲を4分音符から符点4分音符の音符長に変更した例を示している。こうして、操作者は、編集すべき波形の範囲が所望の音符長になったところで最終サンプル位置E Pの入力を確定する。そして、ステップS b 4に進むと、操作者の指示に応じて上記第1実施例と同様の編集処理①〜⑦を行う。

【0032】このように、本実施例によれば、所望の時点からの編集すべき波形の範囲の指定を算出される音符長を参照しながら確定することができるので、上記第1実施例と同様、収録した音楽の波形データを編集する作業を容易に行うことができる。

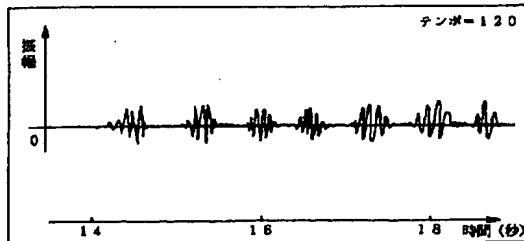
【0033】なお、上記第1および第2実施例において波形データの位置指定の単位とした波形の1サンプル（波形記録時のサンプリング周期に対応）は、必ずしもRAM 3上の1アドレスに対応していなくともよい。例えば、サンプル数「3」が2アドレスに対応している場合、両者の対応関係に応じたアドレス計算を行えばよいことになる。

【0034】また、上記第1および第2実施例では、RAM 3上の領域をシリアルに使用することを想定して説明したが、本発明はこれに限らず、領域を不連続に使用する場合（例えば、RAM 3上の領域を複数のブロックに分割し、これらブロックを任意に連結して使用する場合）にも適用可能である。この場合においても、各領域（ブロック）の連結状態に応じたアドレス計算を行えばよいことになる。また、再生速度を変更する（時間軸圧縮伸長技術により速度を変えたり、単にサンプルの読み出し速度を変える）ときには、指定するテンポもそれに応じて変更すればよい。こうすることにより、同一音符長のサンプル数が変更前と共通になるので、編集作業が楽になる。

【0035】

【発明の効果】以上説明したように、請求項1記載の発

【図2】



8

明によれば、編集処理を施すべき波形データの範囲を音符長で指定することができる。また、請求項2記載の発明によれば、編集処理を施すべき波形データの範囲を指定する際に、該範囲の長さを音符長で知ることができる。これにより、音符の長さを意識した波形データの編集作業が容易に行えるという効果が得られる。

【図面の簡単な説明】

【図1】 この発明の第1実施例の全体構成を示すブロック図である。

10 【図2】 同実施例において、モニタに表示される波形の一例を示すグラフである。

【図3】 同実施例において、モニタに表示される波形のグラフの横軸（時間）を小節番号と該小節の先頭からの拍数とで表現した例を示す図である。

【図4】 同実施例において、CPU 1が実行する制御プログラムの編集処理ルーチンを示すフローチャートである。

【図5】 同実施例において、指定された波形の範囲の表示例を示す図である。

20 【図6】 同実施例における波形の削除例を示す図である。

【図7】 同実施例における波形の移動例を示す図である。

【図8】 同実施例における波形のコピー例を示す図である。

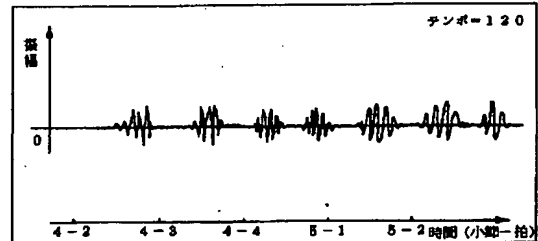
【図9】 この発明の第2実施例において、CPU 1が実行する制御プログラムの編集処理ルーチンを示すフローチャートである。

30 【図10】 同実施例において、最終サンプル位置の入力をし直しにより、編集すべき波形の範囲を4分音符から符点4分音符の音符長に変更した例を示す図である。

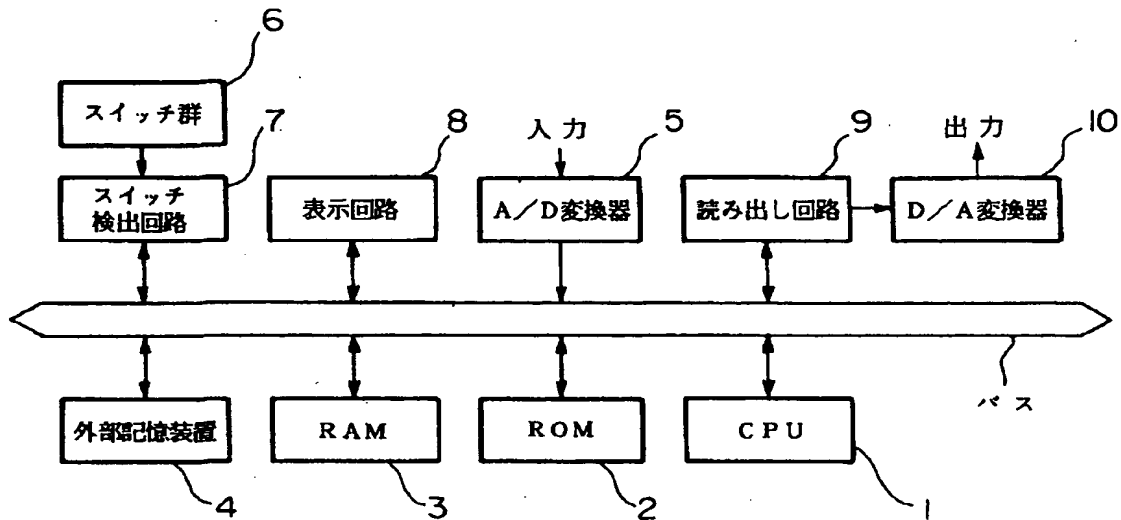
【符号の説明】

1……CPU、2……ROM、3……RAM、4……外部記憶装置、5……A/D変換器、6……スイッチ群、7……スイッチ検出回路、8……表示回路、9……読み出し回路、10……D/A変換器

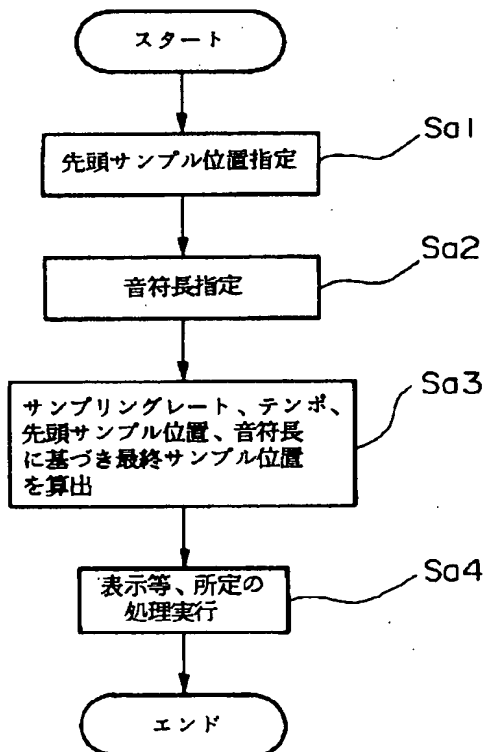
【図3】



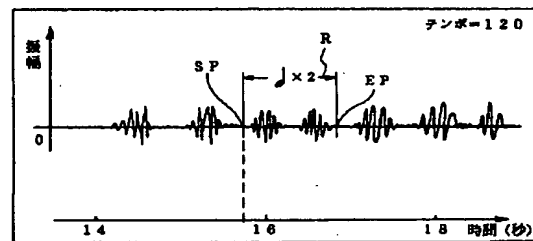
【図1】



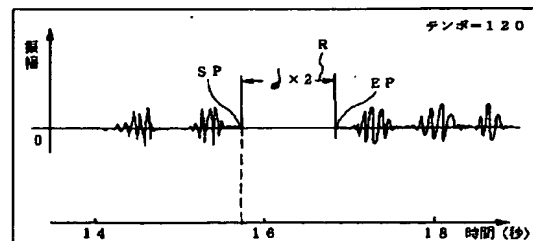
【図4】



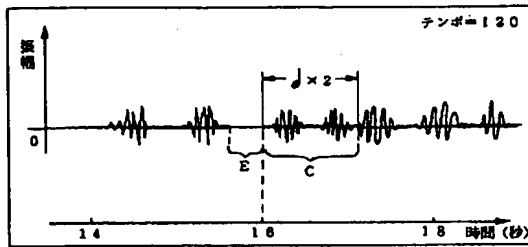
【図5】



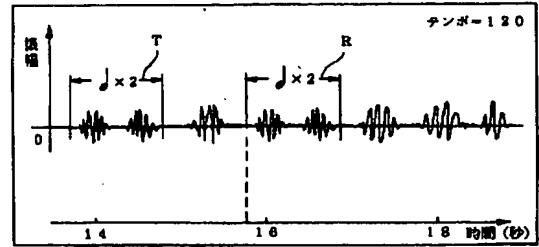
【図6】



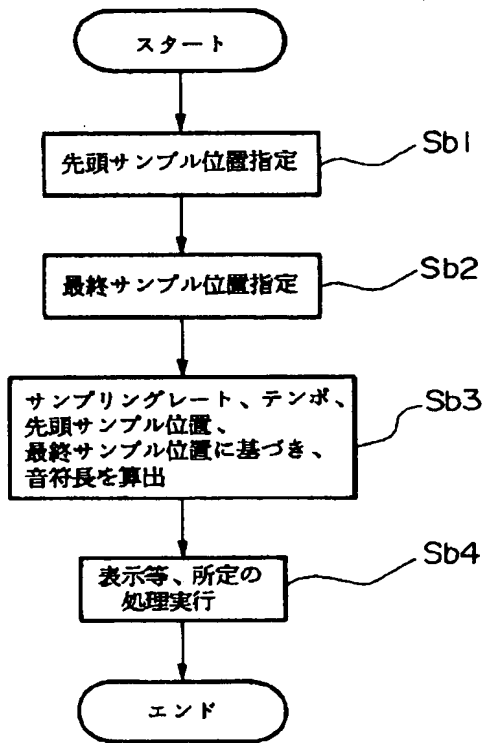
【図7】



【図8】



【図9】



【図10】

